

## IN THE CLAIMS

Please amend the claims as follows where a copy of the claims with the amendments delineated are set forth below in accordance with the PTO guidelines. This listing of claims will replace all prior versions, and listings, of claims in this application.

1. (Previously Presented) A computer implemented method for dynamically rendering data in a markup language, the method comprising:  
identifying a symbol in the data in the markup language, the symbol indicating a query of a data set, the query containing one or more variables, each variable of one of a plurality of data types;  
augmenting the markup language to support the variables;  
accessing the data set in order to generate a resolution to the query;  
substituting the resolution for the query; and  
dynamically rendering the resolution as a part of the markup language, according to at least one rule associated with the markup language wherein said symbol can be used to dynamically render multiple data sets.
2. (Original) The method of claim 1, wherein:  
the symbol comprises a delimited token.
3. (Original) The method of claim 1 wherein:  
the symbol is located within the data in the markup language such that the query is associated with a markup language tag.
4. (Original) The method of claim 3 wherein:  
the markup language comprises Hyper Text Markup Language.

5. (Previously Presented) The method of claim 3 wherein rendering further comprises:  
rendering the resolution according to at least one rule associated with the markup  
language tag with which the query is associated.
6. (Original) The method of claim 1 wherein:  
the data set comprises a set of at least one document in a hierarchically structured format.
7. (Original) The method of claim 6 wherein:  
the hierarchically structured format comprises Extensible Markup Language.
8. (Original) The method of claim 7 wherein:  
the symbol conforms an Extensible Markup Language standard concerning queries.
9. (Original) The method of claim 1 wherein:  
the data set comprises a database.
10. (Canceled)
11. (Original) The method of claim 1 wherein:  
rendering is performed by software running on a hand held computing device.
12. (Original) The method of claim 1 further comprising:  
generating a resolution to the query by retrieving a node set from a set of documents in  
Extensible Markup Language; and  
rendering each member of the node set.
13. (Canceled)
14. (Previously Presented) The method of claim 1 wherein:  
each variable contained in the query comprises a delimited token.
- 15-16. (Canceled)

17. (Original) The method of claim 1 wherein rendering the resolution further comprises:

updating the data set.

18. (Original) The method of claim 17 wherein updating the data set further comprises: writing to a set of at least one document in Extensible Markup Language.

19. (Previously Presented) A computer program product for dynamically rendering data in a markup language, the computer program product comprising:

program code for identifying a symbol in the data in the markup language, the symbol indicating a query of a data set, the query containing one or more variables, each variable of one of a plurality of data types;

program code for augmenting the markup language to support the variables;

program code for accessing the data set in order to generate a resolution to the query;

program code for substituting the resolution for the query;

program code for dynamically rendering the resolution as a part of the markup language, according to at least one rule associated with the markup language wherein said symbol can be used to dynamically render multiple data sets; and

a computer readable medium on which the program codes are stored.

20. (Previously Presented) The computer program product of claim 19 further comprising:

program code for rendering the resolution according to at least one rule associated with a markup language tag with which the query is associated.

21. (Original) The computer program product of claim 19 further comprising:

program code for generating a resolution to the query by retrieving a node set from a set  
of documents in Extensible Markup Language; and  
program code for rendering each member of the node set.

22. (Original) The computer program product of claim 19 further comprising:  
program code for updating the data set.

23. (Original) The computer program product of claim 22 wherein the program code for  
updating the data set further comprises:

program code for writing to a set of at least one document in Extensible Markup  
Language.

24. (Previously Presented) A computer system for dynamically rendering data in a  
markup language, the computer system comprising:

an identification module, for identifying a symbol in the data in the markup language, the  
symbol indicating a query of a data set, the query containing one or more  
variables, each variable of one of a plurality of data types, the identification  
module further adapted to augment the markup language to support the variables;  
a data access module, for accessing the data set in order to generate a resolution to the  
query, the data access module being coupled to the identification module, the data  
access module further adapted to substitute the resolution for the query; and  
a rendering module, for dynamically rendering the resolution as a part of the markup  
language, according to at least one rule associated with the markup language  
wherein said symbol can be used to dynamically render multiple data sets, the  
rendering module being coupled to the data access module.

25. (Previously Presented) The system of claim 24 wherein:

the rendering module is further for rendering the resolution according to at least one rule associated with a markup language tag with which the query is associated.

26. (Original) The system of claim 24 further comprising:

a resolution generation module, for generating a resolution to the query by retrieving a node set from a set of documents in Extensible Markup Language, the resolution generation module being coupled to the data access module; and  
the rendering module is further for rendering each member of the node set.

27. (Original) The system of claim 24 further comprising:

an updating module, for updating the data set, the updating module being coupled to the rendering module.

28. (Original) The system of claim 27 wherein:

the updating module is further for updating the data set by writing to a set of at least one document in Extensible Markup Language.

29. (Original) The method of claim 3 wherein:

the markup language comprises Wireless Markup Language.

30. (New) The method of claim 1 wherein augmenting the markup language to support the variables comprises:

providing a variable table for storing names and values of the variables, each variable of one of the plurality of data types;

utilizing a syntax in the markup language for creating the variables by adding the variables to the variable table; and

building a variable resolution functionality into the markup language, each variable name resolving to one or more corresponding values.

31. (New) The method of claim 1 wherein the one or more variables contained in the query are resolved as part of the generation of the resolution to the query, the query associated with a tag in the markup language.